Remarks:

Applicants appreciatively acknowledge the Examiner's confirmation of receipt of Applicants' claim for priority and certified priority document under 35 U.S.C. § 119(a)-(d).

Reconsideration of the application is respectfully requested.

Claims 1 - 7 are presently pending in the application. Claim 1 has been amended.

In paragraphs 4 - 6 of the above-identified Office Action, there was set forth an objection to the drawings. More specifically, Fig. 2 was objected to as including a typo. That typo has been corrected and a substitute sheet of drawings has been submitted.

The figures were also objected to as allegedly not showing the **offset** of claim 7. Figs. 4A and 4B have been added to show this feature. New Figs. 4A and 4B are fully supported by the present specification in the description extending from page 11, line 13 - page 14, line 7. No new matter is added by the inclusion of new Figs. 4A and 4B. Further, the specification has been amended to make reference to the new Figs. 4A and 4B.

In paragraph 7 of the Office Action, the Examiner objected to the specification because the title was allegedly not

descriptive. The title has been amended to address the present objection.

In paragraphs 8 and 9 of the Office Action, the claims 1 and 2 were rejected as being indefinite under 35 U.S.C. § 112, second paragraph because claim 1 recites the phrase "each case" in line 2. Claim 1 has been amended to delete the phrase "each case" and to better clarify that the claim.

In paragraphs 10 - 12 of the Office Action, claims 3 and 4 were rejected as allegedly being anticipated under 35 U.S.C. § 102(b) by Patterson and Hennessy, Computer Organization, The Hardware/Software Interface 2nd Edition ("**P&H**"). In paragraphs 13 - 27 of the Office Action, claims 1, 2, 5, 6 and 7 were rejected as allegedly being obvious under 35 U.S.C. § 103(a) over U. S. Patent No. 4,821,183 to Hauris ("**HAURIS**") and further in view of **P&H.** Applicants respectfully traverse the above rejections.

Before discussing the prior art in detail, it is believed that a brief review of the invention as claimed, would be helpful. Applicants' claimed invention relates to a microprocessor or method that is useful with **"various assembler codes"**. For example, Independent claims 1, 3 and 4, recite, among other limitations, a microprocessor for processing **various assembler**

codes. Independent claim 5 recites, among other limitations, "providing a plurality of program counter for **various** operating states and **assembler codes**".

In the Office Action, a definition is derived which erroneously equates Applicants' claimed "**various assembler codes**" as being merely "**various basic instructions**". More specifically, in the Office Action, at pages 5 and 6, it is stated:

> "A microprocessor for processing various assembler codes: ('Assembly code' is a synonym for 'assembler code' (IEEE Dictionary) and is defined as 'computer instructions and data definitions expressed in a form that can be recognized and processed by an assembler.' (IEEE dictionary) An assembler is "a program that translates a symbolic version of an instruction into the binary version' (Glossary of Patterson and Hennessy) Figure 5.33 of page 383 in Patterson and Hennessy, discloses a 'Multicycle datapath for MIPS handles the basic instruction' and is further described below figures 5.31 and figure 5.32. An assembler translates various assembly codes into various basic binary instructions for said multicycle datapath microprocessor. **Therefore, a multicycle datapath for MIPS that handles various basic instructions is handling various assembly codes.**" [emphasis added by Applicants]

Applicants respectfully disagree with the derivation and the definition of "**various assembler codes**" set forth in the Office Action. In the present application, Applicants have, themselves, defined what is meant by the terms "**assembler code**" and "**various assembler codes**". Page 1 of the present application, states:

> "To allow them to be processed by a microprocessor, computer programs must be translated into what is know as an **assembler code, i.e. into a programming language** that can be directly executed by the microprocessor. There are currently **various** customary **assembler codes** on the market, **for example JAVA byte code or ECO 2000 Assembler.**" [emphasis added by Applicants]

Contrary to the meaning derived in the Office Action and in light of its definition clearly set forth by the Applicants' in the specification, the claimed "**various assembler codes**" are **various programming/assembler languages**. Applicants' definition is also the ordinary and customary definition. An "assembler code" is understood in the art to be "a computer language". More than one "assembler code" is more than one computer language.

However, even if this were somehow not the customary definition, MPEP section 2111.01 states that the words of a claim must be given their 'plain meaning' **unless** they are defined in the specification. As shown above, the terms "**assembler codes**" and "**various assembler codes**" are, in fact, defined in Applicants' specification. The definition of the above terms is further supported in the present application, page 1, line 23 - page 2 line 2, wherein it states:

> "Prior-art microprocessors have until now always been constructed in such a way that they can only process **a single assembler code**. This is disadvantageous of course, since the computer programs then have to be translated for each processor into the **assembler code**

> respectively to be applied." [emphasis added by Applicants]

To apply the definition derived in the Office Action for "**assembler code**" to the above paragraph of Applicants' specification would yield a ridiculous result, i.e. that prior-art microprocessors could only process a single **basic instruction**. For further support see page 2 of the present specification, lines 16 - 26.

In view of the foregoing, the claimed term "**various assembler codes**" means something different and more specific than "**various basic instructions**" as alleged in the Office Action. For purposes of the present claims, "**various assembler codes**" is clearly defined to mean "**various assembler languages**".

Neither **P&H**, nor **HAURIS**, teach or suggest the use of more than one assembler language. Rather, the **P&H** reference relates only to a single assembler language referred to as the **MIPS assembly language**. See, back inside cover of **P&H**. As such, the **P&H** reference fails to teach or suggest Applicants' claimed "**various assembler codes**". Further, claims 1 - 2 and 5 - 7 require, among other limitations, **a plurality of program counters,** each related to the **various assembler codes**. As **P&H** fails to teach a plurality of assembler codes, it additionally

fails to teach Applicants' specifically claimed plurality of program counters of claims 1 - 2 and 5 - 7.

The **HAURIS** reference likewise fails to teach or suggest the use of Applicants' claimed **various assembler codes**. Rather, **HAURIS** discloses a data processing system including a plurality of control store memories. In **HAURIS**, a first control store memory contains the primary microinstruction group, a second control store memory contains a branched-to subroutine microinstruction group and possibly a third control store memory contains tertiary branched-to subroutine microinstruction groups. However, **nothing** in **HAURIS** teaches or suggests that the microinstructions of the primary group, the branched-to subroutine microinstruction group and/or the tertiary branched-to subroutine microinstruction groups are in different **assembler codes**. Reading **HAURIS**, one can only assume that all three groups of microinstructions are in the same assembler code as the main program. To assume otherwise from the teachings of **HAURIS** would only be possible through impermissible hindsight reconstruction of the present invention.

Further, since **HAURIS** fails to teach a plurality of assembler codes, it additionally fails to teach Applicants' specifically

claimed **plurality of program counters** of claims 1 - 2 and 5 - 7 related to the **various assembler codes**.

It is accordingly believed that none of the references, whether taken alone or in any combination, either show or suggest the features of independent claims 1, 3, 4 and 5. Those independent claims are, therefore, believed to be patentable over the art. The dependent claims are believed to be patentable as well because they all are ultimately dependent on independent claims 1 and 5. As it is believed that the claims were patentable over the cited art in their original form, the claims have not been amended to overcome the references.

In view of the foregoing, reconsideration and allowance of claims 1 - 7 are solicited.

In the event the Examiner should still find any of the claims to be unpatentable, counsel would appreciate receiving a telephone call so that, if possible, patentable language can be worked out.

If an extension of time for this paper is required, petition for extension is herewith made.

Please charge any fees that might be due with respect to Sections 1.16 and 1.17 to the Deposit Account of Lerner and Greenberg, P.A., No. 12-1099.

Respectfully submitted,

Kerry P. Sisselman
Reg. No. 37,237

For Applicants

KPS:cgm

November 22, 2004

Lerner and Greenberg, P.A.
Post Office Box 2480
Hollywood, FL 33022-2480
Tel: (954) 925-1100
Fax: (954) 925-1101

Amendments to the Drawings:

Attached please find two (2) replacement sheets to replace the one (1) sheet of drawings originally filed with the present application. The substitute sheets show original Fig. 1, amended Fig. 2, original Fig. 3 and new Figs. 4A and 4B. Fig. 2 has been amended to correct a typographical error. An annotated sheet showing the change made to Fig. 2 is additionally included..

Attachment: Replacement Sheets

Annotated Sheet Showing Changes

~~1/1~~ 1/2

FIG 1

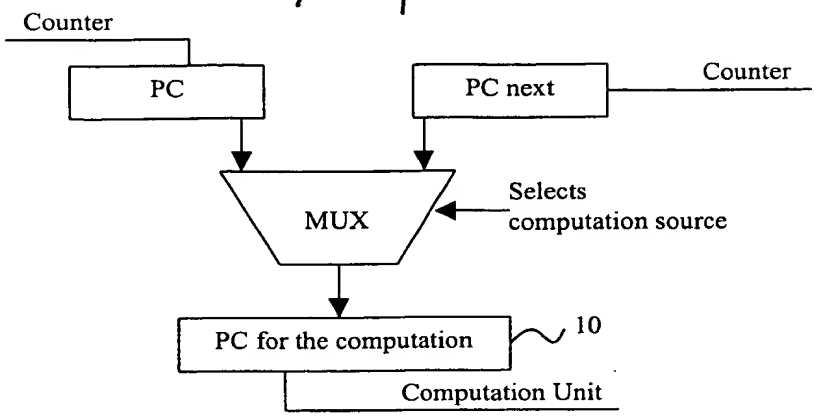

FIG 2

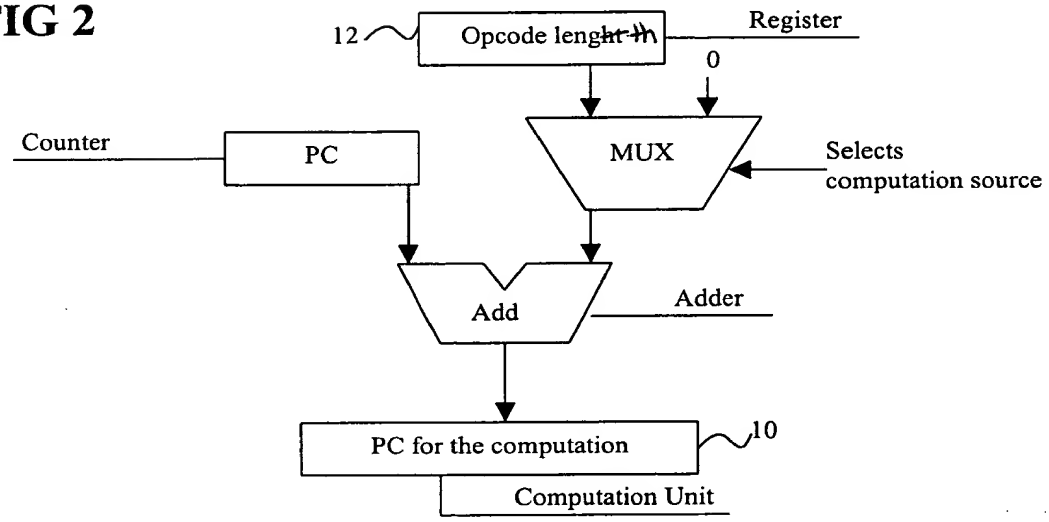

FIG 3